In the Claims:

1. (Original) A method of modeling an electronic system having both hardware and software elements, comprising:

capturing a plurality of behaviors that correspond to operations performed by the system being modeled;

capturing a plurality of hardware and software architectural components the plurality being contained within an architectural platform;

mapping each of the captured behaviors of the plurality of behaviors to a selected architectural component to perform the behavior;

recognizing and capturing communication patterns among the architectural components that require communication among them in order to perform the behaviors; and

mapping each instance of communication between behaviors to an instance of the captured pattern.

2. (Original) The method of Claim 1 wherein each architectural component comprises a plurality of services each service corresponding to a particular function of the architectural component.

3. (Original) The method of Claim 2 wherein each service comprises a declaration and a definition corresponding to the interface and body of the service, respectively.

4. (Amended) The method of Claim 3 wherein each service definition uses other ~~service declarations~~ services.

5. (Original) The method of Claim 1 wherein the step of capturing the architectural platform comprises selecting a performance model for each of the plurality of architectural

components wherein the services of one architectural component uses other services that are topologically connected in the platform.

6. (Original) The method of Claim 1 further comprising a step of partitioning each behavior of the plurality of behaviors that are captured as either a hardware or software behavior.

7. (Original) The method of Claim 1 wherein said capturing communication pattern step comprises a plurality of services each service uses other service declarations that are supported by the architecture component to which the behavior is mapped.

8. (Original) The method of Claim 1 further comprising modifying at least one behavior of the plurality of behaviors or associating at least one behavior with a new appropriate architectural component while maintaining associations among the other behaviors with previously mapped architectural components.

9. (Original) The method of Claim 1 further comprising associating at least one behavior of the plurality of behaviors with a new architectural component and recognizing and capturing one or more new communication patterns while maintaining other previously captured communication patterns.

10. (Original) The method of Claim 1 further comprising changing the selection of one or more pattern mappings while maintaining other previously captured communication patterns.

11. (Original) The method of Claim 1 further comprising modifying at least one architecture component of the plurality of architecture components in the platform while maintaining

associations amongst the other architecture components and patterns of communication to and from the modified component. This modification can be exchange of component for another component or a tweak to the parameter settings or a different selection of performance model.

12. (Original) The method of Claim 1 further comprising modifying the plurality of architecture components that are contained in an architecture while maintaining previously captured behavior mappings and communication patterns.

13. (Original) The method of Claim 1 further comprising at least one member of a group consisting of reuse of services in other architecture components and reuse of architecture components in other architecture platforms or reuse of architecture platforms in other electronic systems.

14. (Original) The method of Claim 1 further comprising reuse of patterns in other electronic systems with variations on the architecture platform.

15. (Original) The method of Claim 1 further comprising simulating operation of the electronic system when performing the plurality of behaviors utilizing the architecture services of the mapped behaviors and optionally the pattern services that were selected.

16. (Original) The method of Claim 15 wherein utilizing the architecture services can be successively refined where additional design decisions are made causing the performance model to become more accurate.

17. (Original) A system for creating a behavioral model of an electronic system having hardware and software components, comprising:

a plurality of architectural components, each of the architectural components corresponding to a component capable of being implemented as part of the electronic system; and

means for generating communication patterns between the architectural components that require communication between them in order to perform user specified behaviors, each communication pattern including communication between any intervening architectural components needed to communicate between architectural components carrying out the behaviors.

18. (Original) The system of Claim 17 wherein each architectural component comprises a plurality of services each service corresponding to a particular function of the architectural component.

19. The system of Claim 17 wherein the means for generating communication patterns between the architectural components comprises the means for selecting the appropriate services of the plurality of services necessary for carrying out communication between the architectural components.

20. (Original) The system of Claim 17 further comprising means for associating each behavior of a plurality of behaviors with an appropriate architectural component.

21. The system of Claim 17 wherein the means for associating each behavior of the plurality of behaviors comprises means for selecting an architectural component model for the architectural component, the architectural component

model being one of a group of predefined architectural component models.

22. The system of Claim 17 further comprising means for classifying each behavior of a plurality of behaviors to be performed by the electronic system as either a hardware or software behavior.

23. The system of Claim 17 further comprising means for associating at least one modified behavior with a new appropriate architectural component while maintaining associations of the other unmodified behaviors with previously mapped architectural components.

24. The system of Claim 17 further comprising means for associating at least one behavior of a plurality of behaviors with a new architectural component.

25. The system of Claim 24 further comprising means for generating one or more new communication patterns between the new architectural component and other architectural components while maintaining other previously generated communication patterns.

26. The system of Claim 17 further comprising a simulation application for simulating operation of the electronic system when performing a plurality of behaviors utilizing the communication patterns that were generated.

27. A performance level model of the communications between behaviors of an electronic system having hardware and software components, the model comprising:

an application programming interface for a first behavior that provides data to be transferred to one or more destination behaviors;

a first service that implements the application programming interface that models the performance of the communication protocol, the first service being among a plurality of services supported by the pattern to which the behavior communication is mapped;

one or more application programming interfaces used by the first service to model performance of the architecture platform, the application interfaces being among a plurality of service declarations supported by the symbol of the architectural component to which the first behavior is mapped;

a supported service declaration on the symbol of the architecture component by a service definition, the service definition being among a plurality of service definitions specified by the performance model of the architecture component;

a second application interface that represents a function to be performed by a second architectural component topologically connected to the first component of the electronic system, the second service being one a plurality of second services each corresponding to a function capable of being performed by the second architectural component;

an input application interface on the destination behavior that receives output information of the performance level model of the electronic system, thereby completing a communication from source behavior to destination behavior.

28. (Original) The performance level model of Claim 27 further comprising a third service that represents a function to be performed by a third architectural component of the

electronic system, the third service being one a plurality of services each corresponding to a function to be performed by the third architectural component and wherein there is at least one application programming interface comprising a first application programming interface that is a model of the communication behavior between the first and second architectural components topologically connected and a second application programming interface that is a model of the communication behavior between the second and third topologically connected architectural components.

29. (New) The method according to Claim 1, wherein:

the architectural blocks include software and hardware architectural blocks; and

the method further comprising the steps of,

re-mapping portions of the behavioral blocks to different architectural blocks and causing implementations of behavior to move between hardware and software.

30. (New) The method according to Claim 29, wherein the step of re-mapping comprises iteratively re-mapping until the modeled system meets specific performance requirements.

31. (New) The method according to Claim 29, wherein said step of re-mapping is performed to alter the communications patterns.

32. (New) The method according to Claim 1, wherein the architectural components are high level architectural components and not component level design items.

33. (New) The method according to Claim 32, wherein at least on the high level architectural components is one of a series of Real Time Operating System (RTOS) scheduling components.

34. (New) The method according to Claim 1, wherein the communication patterns include each of timing, speed and protocols that are required to carry out communication between the separate architectural components.

35. (New) The method according to Claim 1, wherein the step of mapping each instance of communication comprises mapping each instance of communication to a semaphore pattern that includes a sender and receiver pair of pattern services representing each end of the communication.

36. (New) The method according to Claim 35, wherein the sender pattern service models locking of a mutex, writing data, unlocking the mutex, and sending a trigger.

37. (New) The method according to Claim 36, wherein the receiver pattern service models locking the mutex, reading the data, and unlocking the mutex.

38. (New) The method according to Claim 35, further comprising the step of:

selecting a group of patterns from a plurality of pattern groups based on an implementation chosen for the sender receiver pair;

selecting a pattern service to be mapped to each instance of communication from the selected group of patterns;

wherein the plurality of pattern groups include pattern groups for each of ASICs, inter-task software, intra-task software, event trigger transmitting.

39. (New) The method according to Claim 38, wherein the patterns in each groups comprise:

HW - HW	(a) Direct Connect, (b) Register Mapped, (c) Shared Memory
HW - HW Trigger	(a) Direct Connect, (b) Register Mapped
HW - SW	Interrupt Register Mapped, (b) Interrupt Shared Memory, (c) Polling Register Mapped, (d) Polling Shared Memory
HW - SW Trigger	(a) Interrupt, (b) Polling Register Mapped, (c) Polling Shared Memory
SW - HW	(a) Register Mapped, (b) Shared Memory
SW - HW Trigger	(a) Register Mapped
SW - SW Inter-task	(a) Unprotected, (b) Semaphore Protected, (c) Uninterruptable Protected
SW - SW Inter-task Trigger	(a) Unprotected
SW - SW Intra-task	(a) Unprotected
SW - SW Intra-task Trigger	(a) Unprotected
SW->Memory	(a) SWDirectMemoryAccess (b) SWDMAAccess
HW->Memory	(a) HWDirectMemoryAccess (b) HWDMAAccess
SW->Timer	(a) SWVirtualTimer
HW->Timer	(a) ASICInternalTimer

40. (New) The method according to Claim 39, further comprising the step of:

categorizing each of the communications into one of the groups based on mappings of the behavior components and size of data to be communicated.